UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/557,250	04/24/2000	Jonathan S. Goldick	MSFT-0174/150793.1	8456

7590 12/31/2003

Thomas E Watson
Woodcock Washburn Kurtz Mackiewicz & Norris LLP
One Liberty Place
46th Floor
Philadelphia, PA 19103

EXAMINER	
PATEL, HARESH N	

ART UNIT	PAPER NUMBER
2154	

DATE MAILED: 12/31/2003

8

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No. 09/577,250	Applicant(s) IVERSEN ET AL.	
	Examiner Haresh Patel	Art Unit 2154	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☐ Responsive to communication(s) filed on _____.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) 1-27 is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) 1-27 is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. §§ 119 and 120

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

13)☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. § 119(e) (to a provisional application) since a specific reference was included in the first sentence of the specification or in an Application Data Sheet. 37 CFR 1.78.

a) ☐ The translation of the foreign language provisional application has been received.

14)☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. §§ 120 and/or 121 since a specific reference was included in the first sentence of the specification or in an Application Data Sheet. 37 CFR 1.78.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449) Paper No(s) _____.
4) ☐ Interview Summary (PTO-413) Paper No(s). _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: .

DETAILED ACTION

1. Claims 1-27 are presented for examination.

Response to Arguments

2. Applicant's arguments with respect to claims 1-27 have been considered but are moot in view of the new ground(s) of rejection.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

3. Claims 1-10, 12-14, 16-21 and 22-27 are rejected under 35 U.S.C. 103(a) as being unpatentable over Lomet et al. 5,870,763 (Hereinafter Lomet) in view of Luu 6,324,690.

4. As per claims 1, 13, 14, 16 and 22, Lomet teaches the following:

a method for utilizing application dependency information to efficiently perform a backup service operation in a computer system,

a computer-readable medium having computer-executable instructions for instructing a client computer to perform the acts,

a computer system (e.g., database computer system with application recovery and dependency handling read cache, title), comprising:

a plurality of applications loaded in said system (e.g., applications recoverable from system crashes, col. 5, lines 31-38), wherein at least one application has at least one external data dependency associated therewith (e.g., the operation sequence in FIG. 10 introduces a dependency between the application object A and the data object O, col. 17, lines 43-55),

a storage component for storing application dependency information (e.g., the object table includes fields to track dependencies among the objects, col. 6, lines 50-60); an agent that functions according to communication protocols of an application programming interface (API) in said system (e.g., interactions with the resource manager, col. 12, lines 43-50) for processing application dependency information communicated to said API from said agent (e.g., different transitions between application states as the application executes, col. 12, lines 43-50) and for storing the application dependency information in said storage component (e.g., any modifications to the application state are accumulated and the application state is flushed from time to time to stable storage using an atomic write procedure, col. 5, lines 41-63), and

a service for making requests to said agent for a set of application dependency information (e.g., an application calls to the resource manager and the resource manager facilitates the task requested by the application, col. 12, lines 59-65, figure 5), wherein said agent collects, stores and packages said application dependency information in response to a request by a service (e.g., Write graph 144 demonstrates a flush order dependency between the application object and data object, the cache manager flushes the application object represented by node 146, thereby installing the read operation, prior to flushing the data object represented by node 148, col. 19, lines 8-14, figure 12), and delivers said set of application dependency information to said

service for further processing by said service (e.g., the recovery manager retrieves the most recently flushed data objects and application objects in the stable database, col. 31, lines 32-46),

an application programming interface (API) for use in a computer system (e.g., the operation sequence in FIG. 10 introduces a dependency between the application object A and the data object O, col. 17, lines 43-55), wherein said API protocol enables an agent to collect, store and package application dependency information in response to a request by a service, (e.g., an application calls to the resource manager and the resource manager facilitates the task requested by the application, col. 12, lines 59-65, figure 5, Write graph 144 demonstrates a flush order dependency between the application object and data object, the cache manager flushes the application object represented by node 146, thereby installing the read operation, prior to flushing the data object represented by node 148, col. 19, lines 8-14, figure 12), and thereafter delivers said application dependency information to said service for further processing by said service (e.g., the recovery manager retrieves the most recently flushed data objects and application objects in the stable database, col. 31, lines 32-46).

However, Lomet does not specifically mention about the application dependency among applications.

Luu teaches the following:

application dependency information among applications (e.g., The installation package consists of an IPACK format file and the files contained in an application software program. The IPACK format file contains sets of commands that are used to modify system files and perform other functions necessary to the installation of the application software. Further, a personality file may be defined which allows for custom tailoring of the installation on a user's

workstation. Further, a UPACK format file provides instructions for deinstalling application software. Deinstallation of application software is necessary for removing unwanted or outdated applications from the user's workstation, col., 2, lines 14 – 26).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Lomet with the teachings of Luu in order to utilize application dependency information for the efficient backup task.

5. As per claims 2, 18 and 24, Lomet teaches the following:

said backup service includes a snapshot service (e.g., atomic and simultaneous flushing of multiple objects, col. 7, lines 2-7).

6. As per claims 3, 4, 19 and 25, Lomet teaches the following:

said backup service includes a determination of an application freeze order (e.g., attention must be paid to the order in which objects are flushed to stable storage, col.6, lines 41-49),

said backup service includes an execution of the freezing of applications in the order reflected by the determined application freeze order (e.g., attention must be paid to the order in which objects are flushed to stable storage, col.6, lines 41-49).

7. As per claim 5, Lomet teaches the following:

loading said application dependency application programming interface (API) into said computer system (e.g., write graph 144 demonstrates a flush order dependency between the

application object and data object, to ensure correct recovery of the application, the cache manager flushes the application object represented by node 146, col. 19, lines 9-14).

8. As per claims 6, 7, 10 and 27, Lomet teaches the following:

said backup service requesting a set of application dependency information from a common software agent for use in connection with the restore operation (e.g., during recovery, the database computer system can invoke a conventional recover manager to recover the application state and object state at the instance of the crash, the conventional recovery manager retrieves the most recently flushed data objects and application objects in the stable database, col. 31, lines 32-46),

said set of application dependency information is the minimum set of information from said storage component for successfully completing the restore operation (e.g., during recovery, the database computer system can invoke a conventional recover manager to recover the application state and object state at the instance of the crash, the conventional recovery manager retrieves the most recently flushed data objects and application objects in the stable database, col. 31, lines 32-46),

comprising the act of unregistering an application (e.g., during recovery, the database computer system can invoke a conventional recover manager to recover the application state and object state at the instance of the crash, the conventional recovery manager retrieves the most recently flushed data objects and application objects in the stable database, col. 31, lines 32-46).

9. As per claims 8, 17 and 23, Lomet teaches the following:

said agent issuing a request to at least one registered application for information from said set of application dependency information requested by the service (e.g., during recovery, the database computer system can invoke a conventional recover manager to recover the application state and object state at the instance of the crash, the conventional recovery manager retrieves the most recently flushed data objects and application objects in the stable database, col. 31, lines 32-46),

wherein the service to which said agent delivers said information is a backup service. wherein the service to which said agent delivers said information is a backup service (e.g., during recovery, the database computer system can invoke a conventional recover manager to recover the application state and object state at the instance of the crash, the conventional recovery manager retrieves the most recently flushed data objects and application objects in the stable database, col. 31, lines 32-46).

10. As per claims 9, Lomet teaches the following:

at least one registered application communicating information to said agent in response to a request by said agent (e.g., an application calls to the resource manager and the resource manager facilitates the task requested by the application, col. 12, lines 59-65, figure 5), said information relating to said at least one application's external dependencies (e.g., Write graph 144 demonstrates a flush order dependency between the application object and data object, the cache manager flushes the application object represented by node 146, thereby installing the read operation, prior to flushing the data object represented by node 148, col. 19, lines 8-14, figure 12).

11. As per claims 12, 20 and 26, Lomet teaches the following:

said agent stores said application dependency information in a tabular format reflective of hierarchical application dependencies in said storage component (e.g., the cache manager thus keeps an object table which tracks dependencies to ensure a proper flushing order, col. 18, lines 4-12).

12. As per claim 21, Lomet teaches the following:

a data structure for storing application dependency information, comprising:

an identifier identifying an application, for which application at least one external dependency is known (e.g., the object table includes fields to track dependencies among the objects, col. 6, lines 50-60); and

data representative of said at least one external dependency (e.g., the object table includes fields to track dependencies among the objects, col. 6, lines 50-60),

an application programming interface (API) for communications of application dependency information relating to applications in a computer system according to the data structure of claim 14 (e.g., the operation sequence in FIG. 10 introduces a dependency between the application object A and the data object O, col. 17, lines 43-55).

13. Claims 11 and 15 are rejected under 35 U.S.C. 103(a) as being unpatentable over Lomet and Luu in view of Lewis 6,513,019.

Lomet and Luu does not specifically show the limitations of claims 11 and 15.

As per claims 11 and 15, Lewis teaches the following:

said API protocol is XML protocol (e.g., Input data are presented to the system from numerous external sources, via an object-oriented API, the Extended Markup Language (XML) and recognizes messages that conform to industry standard formats (including FIX, ISITC, OFX, S.W.I.F.T., ISO, and the like) as well as proprietary format. this application is used to load balance the workflow, backup and recover components, and the like,

said data representative of said at least one external dependency is stored in XML format (e.g., Input data are presented to the system from numerous external sources, via an object-oriented API, the Extended Markup Language (XML) and recognizes messages that conform to industry standard formats (including FIX, ISITC, OFX, S.W.I.F.T., ISO, and the like).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Lomet and Luu with the teachings of Lewis in order to provide a web standard common middleware layer in a communication stack at the API level between applications.

Conclusion

14. Examiner makes a note that the applicant needs to specifically clear the term "application dependency information among applications". Claim 14 does not mention about the dependency among applications with an agent executing on the system.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Haresh Patel whose telephone number is (703) 605-5234. The examiner can normally be reached on Monday-Friday from 8:00 am to 5:30 pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, John Follansbee, can be reached at (703) 305-8498.

The appropriate fax phone number for the organization where this application or proceeding is assigned is (703) 306-5404.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is (703) 305-3900.

Haresh Patel

December 28, 2003

JOHN FOLLANSBEE
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100